Vision Based Navigation for Debris Removal Missions

Keyvan Kanani
Astrium Satellites, France, keyvan.kanani@astrium.eads.net

Antoine Petit[1], Eric Marchand[1], Thomas Chabot[2] and Bernard Gerber[2]

In-orbit servicing, and especially large debris removal, holds more and more the attention of space community. In this paper we present a vision based solution which enables to assess the pose (attitude and position) of a target using a monocular camera. The proposed image processing algorithm achieves 3D detection and tracking of the target, relying on frame-to-frame tracking and tracking by detection techniques. The main principle is to align the projection of the 3D model of the tracked debris with observation made in the image and provides the complete relative position and attitude thanks to non-linear optimization process. Running at video rate, the proposed algorithm can handle very complex targets, under degraded conditions, i.e. with Earth background, occlusion or optical peculiarities (such as poor texture or specular reflections).
The tracking algorithm has previously been tested on real images to demonstrate its robustness to real data. In the present paper we focus on heavy debris tracking, like satellites or launcher tanks. Several tests on simulated images have been carried out to quantitatively show the performances and robustness of both tracking and associated navigation.

I. Introduction

The active removal of heavy space debris (typically larger than 1000kg) has been identified as a key development to control the growth in the debris population and to limit the risk for active satellites. European Space Agency as well as NASA proposes several studies or experiments dealing with debris removal or more generally to orbit servicing (Orbital Express experiment [1], CleanSpace program [2]…). In that context, Astrium Satellites has been working on optimization and implement of sensors and navigation solutions onboard a Debris Removal Vehicle named "The Debritor" with the main objective to ensure high safety proximity maneuvers. In particular, special attention has been paid to the design of autonomous, vision-based navigation solution for uncooperative rendezvous with space debris. The proposed solution enables the estimation of chaser states with respect to

[1] INRIA-Lagadic, France
[2] Astrium Satellites, France

debris, using a single camera (no active sensors or stereo device are considered) and knowing the 3D model of the target, which can be cooperative or not. If this solution has been developed for debris removal, it is applicable to larger field within the in-orbit services domain.

In order to estimate the pose of a camera with respect to a specific scene, common model-based approaches use either point [3], edge features [4,5,6] or a combination of both [7,8]. Edge features offer a good invariance to illumination changes or image noise and are particularly suitable with poorly textured scenes, whether the scene is in industrial, outdoor or indoor environments. For such class of approaches, the pose computation is achieved by minimizing the distance between the projected edges of the 3D model and the corresponding edge features in the image, extracted thanks to a 1D search for gradient maxima along the model edge normals.

Solution presented in this paper has been developed by INRIA and specialized to rendezvous problematic in collaboration with Astrium Satellites. It belongs to edge based trackers. It aims at fulfilling several challenges, in term of accuracy, robustness to on-orbit conditions (Earth background, directional illumination, constrained trajectories…) and on-board implementation (real-time especially).

In a first section, the solution is described. In a second, typical performances are presented.

II. 3D model based pose estimation

Proposed solution is divided into two steps: initialization and tracking. Both steps use the known 3D model of the target. This model can be the CAD model or a simplified one.
This section provides the outline of the solution. For a more detailed description, please refer to [9].

II.1. Initialization

Initialization aims at detecting the target in an image sequence and at providing the tracking with an initial guess of the target pose, without any prior information on the pose. It consists in matching (detection/matching stage) the image contours with a database of views built before the mission (learning stage).

Offline learning. A hierarchical model view graph leading to prototype views *Vj* of the model is built. Each node of the view graph contains an image projection of the target contours at a particular point of view. The points of view are sampled on a sphere (see Figure 1). The sampling of the views is optimized to limit the memory size of the database and to insure the whole coverage of the space of possible views.

Online target detection. Silhouette of the target is extracted in the image using bilayer segmentation techniques. This method consists in minimizing an energy function combining motion and color, along with temporal and spatial priors. It allows distinguishing the foreground shape from the background and has the advantage to be real-time.

Online matching and pose initialization. The view graph is then explored to find the prototype view whose contours correspond the most to the extracted silhouette. The used similarity metric derives from [10]. It considers both the distance and the orientation of edges to match:

$$s(i/j) = \frac{1}{M}\sum_{k=1}^{M} d_j\left(c_k^i\right) + \lambda d_j^{ori}\left(c_k^i\right)$$

With

$$d_j\left(c_k^i\right) = \frac{\min_{l\in[0\ldots n]}\left\|c_k^i - c_l^i\right\|_2}{\left\|c_k^i - cog^i\right\|_2}$$

$d_j\left(c_k^i\right)$ gives the mean distance for each contour point c_k^i to the closest one in prototype view V_j. The metric is normalized by the distance between c_k^i and the centroïd cog^i of the shape extracted from V_j to cope with scale changes.

$$d_j^{ori}\left(c_k^i\right) = \left|ori\left(c_k^i\right) - ori\left(c^j_{\arg\left(d_j\left(c_k^i\right)\right)}\right)\right|$$

This distance stands for the difference between the orientation *ori* of the contour point c_k^i and the orientation of the closest contour point $c^j_{\arg\left(d_j\left(c_k^i\right)\right)}$ in V_j.

Once the closest prototype view is found, its associated pose is considered as initialization of the target pose.

The matching stage can be rather time consuming. To cope with real time, a Bayesian framework is set to spread the initialization over several images (temporal initialization). It enables to provide an up to date pose initialization to the GNC system.

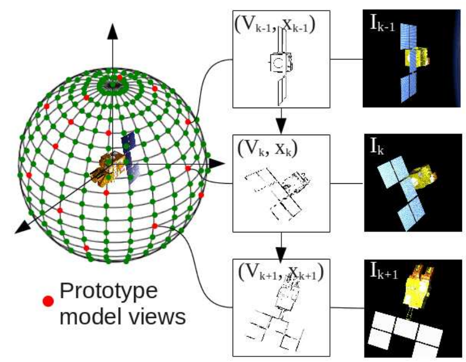


Figure 1 – Principle of initialization. *Several views on a sphere are selected to produce prototype views stored in a hierarchical model view graph. Target is then extracted by segmentation and matched over successive frames with closest prototype.*

II.2. Tracking

Once the target has been detected in image, and its pose has been initialized, a frame to frame edge tracking is performed.
Like initialization, tracking is 3D model based. It aims at finding the target pose which makes best match the projection of the 3D model with the image edges. Tracking and pose estimation are thus simultaneous. They are performed in a fast iterative way, by minimization of following criteria:

$$\Delta = \sum_i \rho\left(d_\perp\left(l_i(r), x_i'\right)\right)$$

where ρ is a robust estimator used to reject outliers (Tuckey estimator), and $d_{\perp}(l_i(r), x_i')$ is the distance between a point x_i' and the corresponding line $l_i(r)$.

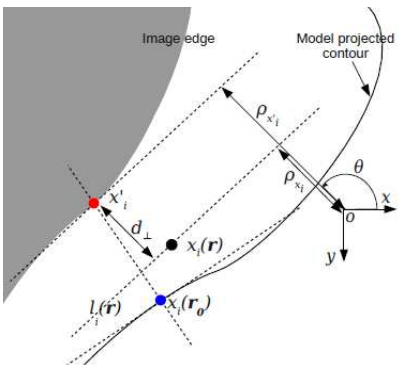


Figure 2 - Moving edge principle. *From the initial pose r0, 1D search along the projected contour underlying the measurement point. Distance of a point x0i to a corresponding line li (r) within the minimization process.*

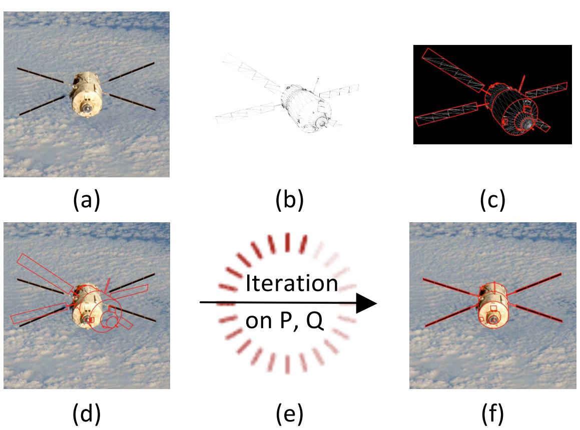


Figure 3 – Principle of tracking. *Tracking is performed using a frame (a) and the 3D model of the target (b). The salient edges of the target are extracted (c) and projected into image (d), given an initial pose. Pose is iteratively refined (e) to make projection edges match with image edges (f).*

Unlike initialization, the edge matching is local. As a consequence, tracking runs in real time but is less robust to high differences between edges, meaning that predicted target pose shall be close enough (tens of pixels) to real one.

III. Performance assessment

Presented solution has been previously tested on satellite mockup and real images [11]. These tests enabled to demonstrate that target detection and tracking work on real images, but they did not enable to assess quantitative performances, as no ground truth was available. To fill the gap, proposed solution has been tested on synthetic images. The results of these tests are presented in this section.

III.1. Image simulation

Synthetic images of debris are simulated using an in-house tool called Surrender!. This renderer is based on classical rendering functions (rasterization, ray-tracing…), adapted to peculiarity of space environment (few objects, long distances) and debris properties (MLI texture, high specularity of surface…). Surrender! is coupled with a simplified space dynamics simulator to simulate trajectories and satellite dynamics sufficiently realistic for testing purpose.

For the performance assessment, two targets are simulated: Ariane 4 upper stage and a Spot family satellite. Typical close range distances, from 100m to 10m, and tumbling rate, from 0 to 2 deg/s (typical rate of large debris), are considered. These parameters have been chosen as they correspond to potential future mission of debris removal.

III.2. Tests results

Miscellaneous tests have been carried out to assess the performances of the solution and its robustness to different conditions (presence of background, target complexity…).
In this section we focus on three representative tests: two on Spot and one on Ariane 4 upper stage.

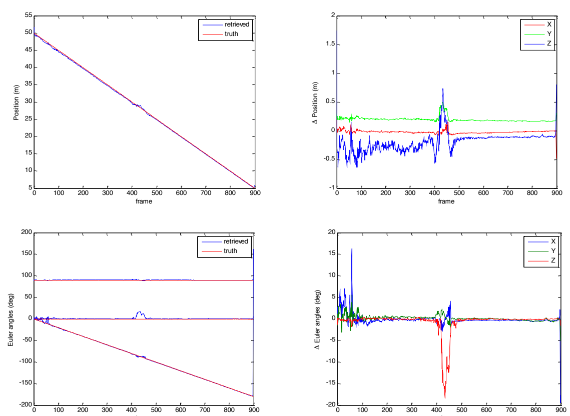
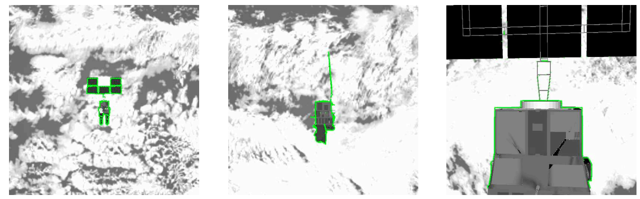


Figure 4 – Spot approach – pose estimation*. (top left) Estimation of z coordinate (line of sight) of target in camera frame. (top right) Error on position. (bottom left) retrieved Euler angles. (bottom right) error on Euler angles.*

First test consists in a simple translation towards a rotating Spot (2deg/s), with the Earth in background, as shown on Figure 5. It enables to provide typical performances of tracking algorithm.



Frame 0 Frame 440 Frame 875

Figure 5 – Spot approach - images. *Green contours correspond to projected 3D model. If contours are aligned with spot contours in image, pose is well estimated.*

Figure 4 shows the estimated pose in comparison to ground truth. Error on position decreases proportionally to distance to target, from 50cm to 5cm, for a distance ranging from 50m to 5m. It shows, as expected, that error on position is directly proportional to image resolution.
Estimation error on attitude is more constant, varying mainly between ±0.5deg, with a minimum error reached when the satellite occupy the whole image, without spilling out it.
First test arises also a hard configuration, when solar arrays and Spot body are perpendicular to image plane (see frame 440 of Figure 5 and associated errors on Figure 4).

In a second test, a rotating Spot (2deg/s) is tracked during a long orbit period. Figure 6 illustrates the pose initialization of this test. Image is firstly segmented and iteratively fitted with prototype views of hierarchical graph. The initial pose corresponds to the pose of the closest prototype view.

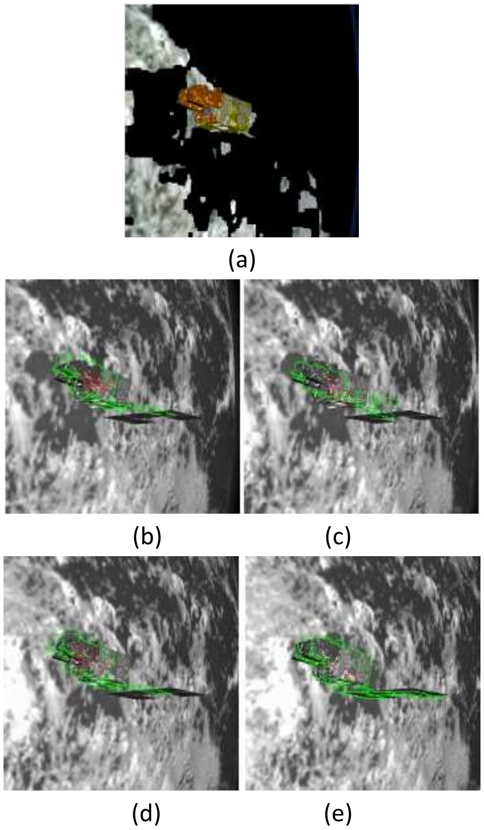


Figure 6 – Spot long tracking - initialization. *Image segmentation on color and motion is performed (a). Matching with prototype views of view graph is then processed on several consecutive frames (b to e).*

Second test shows that tracking can be robust over a long tracking sequence, where sun illumination and background (Earth or not) vary a lot (see Figure 7).

Performances of pose estimation are shown in Figure 8. Errors on pose estimation are quite the same as for test one: position estimation around 50cm for a target between 20m and 70m (from 400px to 150px) and attitude error around 1deg. Pose estimation is erroneous when point of view is close to the hard configuration described for first test.

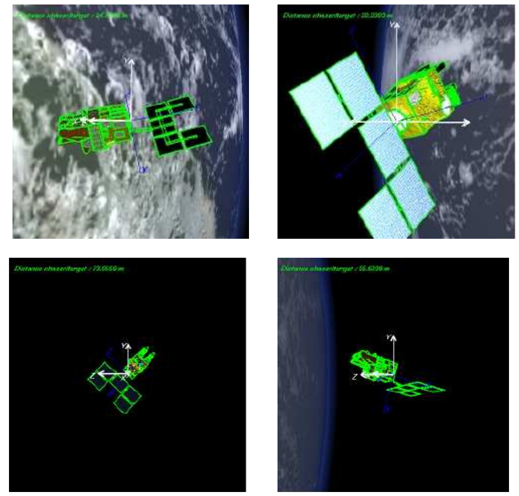


Figure 7 - Spot long tracking - images. *Green contours correspond to projected 3D model*

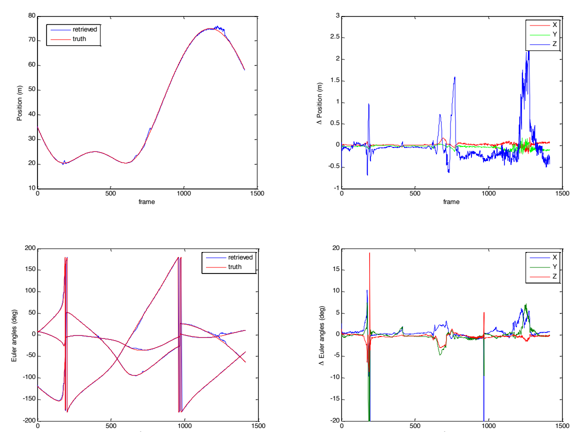


Figure 8 - Spot long tracking pose estimation*. (top left) Estimation of z coordinate (line of sight) of target in camera frame. (top right) Error on position. (bottom left) retrieved Euler angles. (bottom right) error on Euler angles.*

A third test considers Ariane 4 upper stage instead of Spot satellite. Tracking performances are quite the same demonstrating that our solution is robust to target shape. Figure 9 shows an example of upper stage images. Even if texture of stage surface should differ from simulated images, this test makes us confident on the ability of our solution to track launcher upper stages.

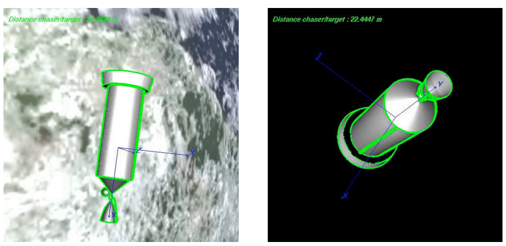


Figure 9 – Ariane 4 upper stage tracking.

III.3. Computational costs

Our solution is coded in C++ and uses GPU functionalities. On a standard computer, the whole algorithm (initialization + tracking) is typically processed at around 15 fps.
Nevertheless, computation cost depends highly on the 3D model complexity, the image size and the inter-frames motion.

IV. Conclusion

In this paper, a vision based solution to assess the pose (attitude + position) of an object in the camera frame has been presented. The solution needs monocular camera images and the 3D model (complete or simplified) of the object as input.

Performances

Several tests have been carried out on simulated images. They consider Spot satellite and Ariane 4 upper stages as targets. Tests show the good performances of pose estimation. Typical an error on position is around 1% of the distance to target (from 100m to 5m with 512x512 images) and an error on attitude less than 1deg.

Operational application

The implementation of the solution is at an advanced prototype level and could be easily transferred to a ground-segment to assess offline (but in real time) the pose of space debris observed by an on-orbit vehicle.
On board and close-loop implementation would need more work, but is seriously foreseen by Astrium Satellites for future debris removal or other in-orbit servicing missions.

References

[1] R. B. Friend. Orbital Express Program Summary and Mission Overview, Proceedings of SPIE, vol. 6958, (2008), pp. 695803-1-695803-11.

[2] Clean Space website: http://www.esa.int/TEC/Clean_Space/

[3] G. Bleser, Y. Pastarmov, and D. Stricker. Real-time 3d camera tracking for industrial augmented reality applications. Journal of WSCG, pages 47–54, 2005.

[4] D.G. Lowe. Fitting parameterized threedimensional models to images. IEEE Trans. on Pattern Analysis and Machine Intelligence, 13(5):441–450, May 1991.

[5] T. Drummond and R. Cipolla. Real-time visual tracking of complex structures. IEEE Trans. on Pattern Analysis and Machine Intelligence, 24(7):932–946, July 2002.

[6] A.I. Comport, E. Marchand, M. Pressigout, and F. Chaumette. Real-time markerless tracking for augmented reality: the virtual visual servoing framework. IEEE Trans. on Visualization and Computer Graphics, 12(4):615–628, July 2006.

[7] L. Vacchetti, V. Lepetit, and P. Fua. Combining edge and texture information for real-time accurate 3d camera tracking. In ACM/IEEE Int. Symp. on Mixed and Augmented Reality, ISMAR'04, pages 48–57, Arlington, VA, November 2004.

[8] E. Rosten and T. Drummond. Fusing points and lines for high performance tracking. In IEEE Int. Conf. on Computer Vision, volume 2, pages 1508–1515, Beijing, China, 2005.

[9] A. Petit, E. Marchand, K. Kanani. Vision-based Detection and Tracking for Space Navigation in a Rendezvous Context. International Symposium on Artificial Intelligence, Robotics and Automation in Space (i-SAIRAS), September 2012

[10] Christian Reinbacher, Matthias Ruether, and Horst Bischof. Pose estimation of known objects by efficient silhouette matching. In 20th International Conference on Pattern Recognition (ICPR), 2010.

[11] A. Petit, N. Despré, E. Marchand, K. Kanani, F. Chaumette, S. Provost, G. Flandin. 3D Model-based Visual tracking for Space Autonomous Rendezvous. 8th International ESA Conference on Guidance & Navigation Control Systems. 2011